

CA2 ON
EV.506

....
R65
C.2

OPTIMIZATION OF ENERGY ALLOCATION IN WATER AND WASTEWATER TREATMENT SYSTEMS

Research Report #W65

March 1977



Ontario

Ministry
of the
Environment

The Honourable
George A. Kerr, Q.C.,
Minister

K.H. Sharpe,
Deputy Minister

Copyright Provisions and Restrictions on Copying:

This Ontario Ministry of the Environment work is protected by Crown copyright (unless otherwise indicated), which is held by the Queen's Printer for Ontario. It may be reproduced for non-commercial purposes if credit is given and Crown copyright is acknowledged.

It may not be reproduced, in all or in part, for any commercial purpose except under a licence from the Queen's Printer for Ontario.

For information on reproducing Government of Ontario works, please contact ServiceOntario Publications at copyright@ontario.ca

CARON
EV. 506
....
R65
c.2

OPTIMIZATION OF ENERGY
ALLOCATION IN WATER
AND WASTEWATER TREATMENT SYSTEMS

By

G.D. Zarnett

Applied Sciences Section
Pollution Control Branch
Publication No. W 65

March 1977

Ministry of the Environment
135 St. Clair Ave. W.
Toronto, Ontario

ABSTRACT

Because reduction of energy has become a major requirement for water and wastewater treatment systems design, introduction of an energy criterion has increased the number of design variables and decisions to be made for plant design and evaluation. To optimize a plant consisting of many stages and recycles, a means of evaluating performance with respect to effluent quality, operating cost and energy must be utilized. This report suggests that the technique of "dynamic programming" be applied as a means of system optimization. Several examples are used to illustrate the application of this optimization method.

TABLE OF CONTENTS

	<u>Page No.</u>
1. Introduction	1
2. Multistage Decision Procedures	1
3. Dynamic Programming	2
4. A Multistage Optimization System	5
5. Minimization of Pipeline Pumping Energy for Water Transmission Systems	12
6. Multi-Unit Water Treatment Plant	14
7. Discussion and Recommendations	17
References	19
Appendix I	20

LIST OF FIGURES

<u>Figure No.</u>	<u>Page No.</u>
1.	3
2.	3
3. 10 mgd Wastewater Treatment System	6
4. Process Alternatives	8
5. Initialization of Dynamic Programming Procedure	9
6. Evaluation of Trajectories	10
7. Optimal Trajectory Path	11
8. Pipeline Pressure Drop - Distance	13
9. Multi-Unit Water Treatment Plant	16

APPENDIX

<u>Figure No.</u>	<u>Page No.</u>
A1	21
A2 Two Steps in Dynamic Programming Example	23
A3 Three Steps in Dynamic Programming Example	23

1. Introduction

The problems associated with energy and energy resource systems which have risen in the past two years have led to intensive efforts to reduce energy consumption. Water and wastewater treatment plants are now being scrutinized for means of reducing energy demands and losses. Earlier investigations by the Applied Sciences Section of the Pollution Control Branch produced a number of reports on energy requirements for water and wastewater systems involving overall energy requirements for treatment systems as well as the requirements for individual unit operations and processes (Zarnett, 1975, 1976, 1976a). The data from these studies can be utilized for design purposes in evaluating alternatives for new plants in which energy expenditure is to be minimized.

In previous years, the general design problem involved minimizing plant capital cost and operational cost and, at the same time, achieving the required effluent standards for which the system was designed. Now that energy reduction has become a significant goal, a means of plant evaluation is necessary in which cost, effluent requirements and energy requirements are optimized. This added dimension in design and planning requires a multitude of new decisions to be made along with a means to evaluate them. The purpose of this proposal is to suggest a means of systematically making decisions which will lead to an optimal energy policy.

2. Multistage Decision Procedure

Any planning problem, be it water treatment plant design, manufacturing plant layout, corporate expansion program, or family vacation trip, involves many decisions, each affecting the other. In planning problems, the components may be arranged in series, each representing the same technological complex at a different point in time. This serial

structure, in which decisions at any stage affect the conditions of all later ones, occurs also in processes having manufacturing units in series. Several systems lend themselves to decomposition by what is termed partial optimization. The earliest and best-known example of the partial optimization approach is Bellman's "dynamic programming" (Bellman, 1957). "Dynamic" is used in the sense of being novel, and "programming" is used to describe the technique as being procedural or algorithmic, and is not related to computer programming, although the method, due to its algorithmic nature, can utilize a computer very effectively.

This method has the capacity to resolve problems in optimization which have "real world" constraints applied to them which were formerly ignored or unsolved, and to attack problems involving multidimensional decisions.

3. Dynamic Programming

Dynamic programming embraces the area of multistage decision processes. A multistage decision process is one, as the name implies, with many stages involving decisions. In particular, in dynamic programming, the decisions must be made in the light of the entire process rather than in the light of each stage as an entity. This means the decision at each stage must be "right" in view of the entire process.

The main concept of this technique lies in the "principle of optimality" (Bellman, 1957):

An optimal policy has the property that, whatever the initial state and the initial decision, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

This principle of optimality is an intuitive concept and can be used as such with no mathematics beyond arithmetic or can be involved in solving complex mathematical models. Early research has shown that this intuitive

form can be reduced rigorously to an exact mathematical form such as the Euler-Lagrange equation of calculus of variations, or to Pontryagin's maximum principle (Bellman, 1962; Lapidus & Luus, 1967).

If optimization is to be performed, it should be regarded as a multistage decision process in which at each stage the control variable y_n is to be chosen in terms of the current state denoted by x_n , and the number of stages remaining in the process. This involves abstracting the mathematical content of the fundamental engineering principle of feedback control. The influence exerted upon the system is dependent upon the current stage of the system. This is an example of event orientation.

If we start in state $x_0 = c$ and choose the control vector y_0 , as in Figure 1, and define a criterion function $J = h(x_n, y_n)$, the result of decision y_0 is $h(x_0, y_0)$.

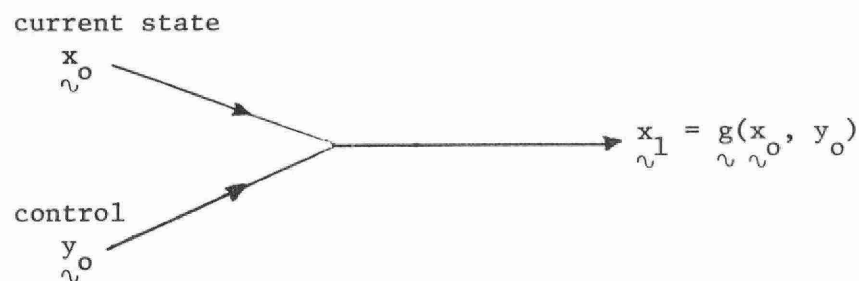


Figure 1

Starting with the new state x_1 , we choose y_1 ; see Figure 2. The new contribution to J is $h(x_1, y_1)$.

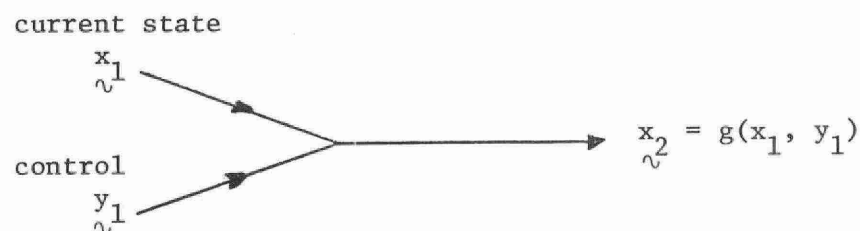


Figure 2

This may be considered to be a multistage decision process with an important invariant structure. Due to the additivity of the criterion function at each stage, the past (both the past history of the process and the past costs) may be ignored when it comes to making present and future decisions. We are free to concentrate on choosing the remaining variables, so as to minimize the remaining part of the criterion function, by starting with a knowledge of the current state.

It is this sharp decomposition into a past, present and future which permits obtaining an analytical hold on the optimization process.

Policies

A decision made in state C, when there are N stages remaining, is written as $y(C,N)$. In general, a function $y(C,N)$ which yields an admissible choice of y_0 is called a policy. A policy which yields the absolute minimum (or maximum) of the criterion function is called the optimal policy. Policy is a basic concept of dynamic programming.

The Performance Index

To measure the "goodness-of-control" in some meaningful manner, it is necessary to introduce a criterion which specifies how well the system is doing. In alternative terms, we wish to specify how close the control achieves any goal which is set up for it. This is accomplished by means of a scalar performance index. A typical example of a performance index is the air pollution index. If the pollution index is exceeded some control is usually applied to reduce or minimize the performance index.

For water and wastewater treatment design, the performance index should include operating costs, degree of contaminant removal and energy requirements. Each of these parameters should be weighted against each other for a realistic design criterion. This type of performance index

can be easily incorporated into the dynamic programming formulation.

A more complete discussion may be found elsewhere, (Athans & Falb, 1966).

4. A Multistage Optimization System

A simple example is shown schematically in Figure 3. A waste-water treatment system is to be designed on the basis of a minimum of energy only, with no consideration of advantages or disadvantages of the unit operations and processes involved or any government regulations or guidelines. The purpose of this simplified example is to illustrate the principle of optimality. The numbers shown with each operation is the performance index which in this case is only the unit energy requirement in kwh/d. Taking the final effluent as the origin = 0, the analysis proceeds stagewise backwards summing the performance indices at each stage. At the origin, $I = 0$; at stage (1) from the right, $I = 303, 935, 462$.

$$\Sigma \min I(0,1) = \min (303, 935, 462) + I(0,0) = 303 + 0 = 303$$

$$\text{For stage (2), } \Sigma \min I(0,2) = \min (13410, 333) + \Sigma I(0,1) = 333 + 303 = 636$$

$$\begin{aligned} (3), \Sigma \min I(0,3) &= \min (13410, 6337, 4368) + \Sigma I(0,2) \\ &= 4368 + 636 = 5004 \end{aligned}$$

$$\begin{aligned} (4), \Sigma \min I(0,4) &= \min (13410, 1150, 5743, 3666) + \Sigma I(0,3) \\ &= 1150 + 5004 = 6154 \end{aligned}$$

$$\begin{aligned} (5), \min \Sigma I(0,5) &= \min (370, 50, 1000, 20000, 375, 75) + \Sigma I(0,4) \\ &= 50 + 6154 = 6204 \end{aligned}$$

$$(6), \min \Sigma I(0,6) = 4.93 + \Sigma I(0,5) = 6209$$

Therefore, the minimum energy requirement is 6209 kwh/d, and the plant should consist of screening, sedimentation, stabilization basin, ammonia stripping, chemical precipitation and ozonation.

This is a simplified example which can be done mentally, but illustrates the technique of dynamic programming. A more complicated

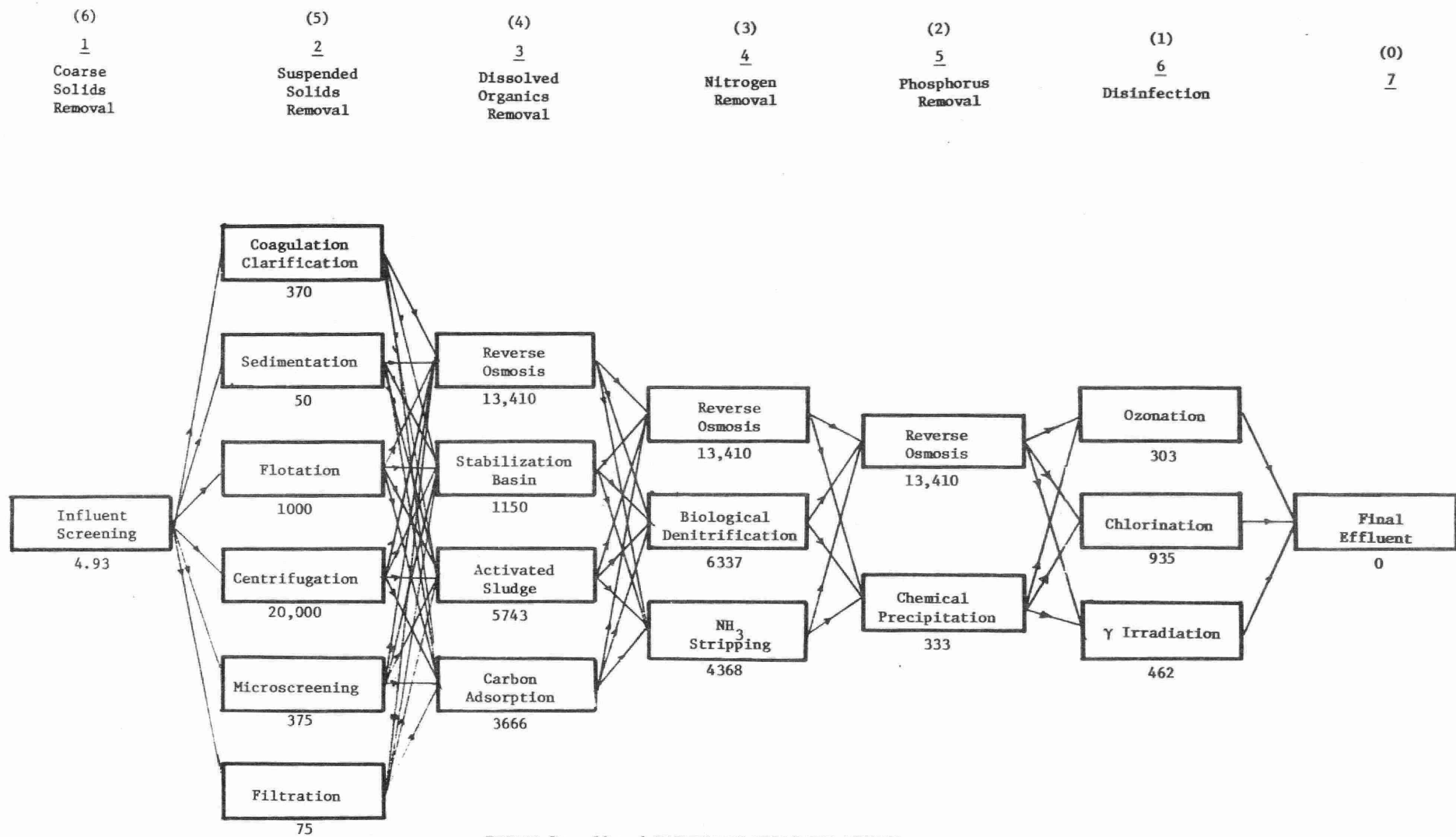


FIGURE 3 10 mgd WASTEWATER TREATMENT SYSTEM

process is shown in Figure 4 in which pumping and auxilliary energies are included in the performance index along a path. This is a 9 stage problem with up to six alternatives and up to three decisions per stage. This problem can be solved by considering it as a trajectory problem in which it is desired to move from A to B while minimizing the performance index to give the shortest path. At each junction designated by a circle, the alternatives are a motion upward to the right, downward to the right or horizontally to the right. Starting at the final point B as the origin, there are four possible preceding states, and the value of the performance index is indicated next to the arrow. To go from stage 0 to stage 1, we must find f_1 for each decision and enter them in the circles. From stage 2 to stage 1, a decision is made for each node by minimizing f_2 at each node and entering the value at the node. This step is shown in Figure 5. The complete analysis is shown in Figure 6 and the optimal trajectory path in Figure 7. The minimum value of the performance index for this path is 166.

Utilizing the principle of optimality, one obtains a considerable reduction in the number of calculations necessary compared with complete enumeration of the possibilities. For a system with three choices at each junction as in the example, the number of calculations for dynamic programming is:

$$[N^3/3 + (N-2)]/2$$

To enumerate all possible paths,

$$\frac{(N-1) (N!)}{(N/3!) (N/3!)}$$

calculations are necessary. A 9 stage problem requires 125 calculations for dynamic programming compared with 80,640 for enumerating all paths.

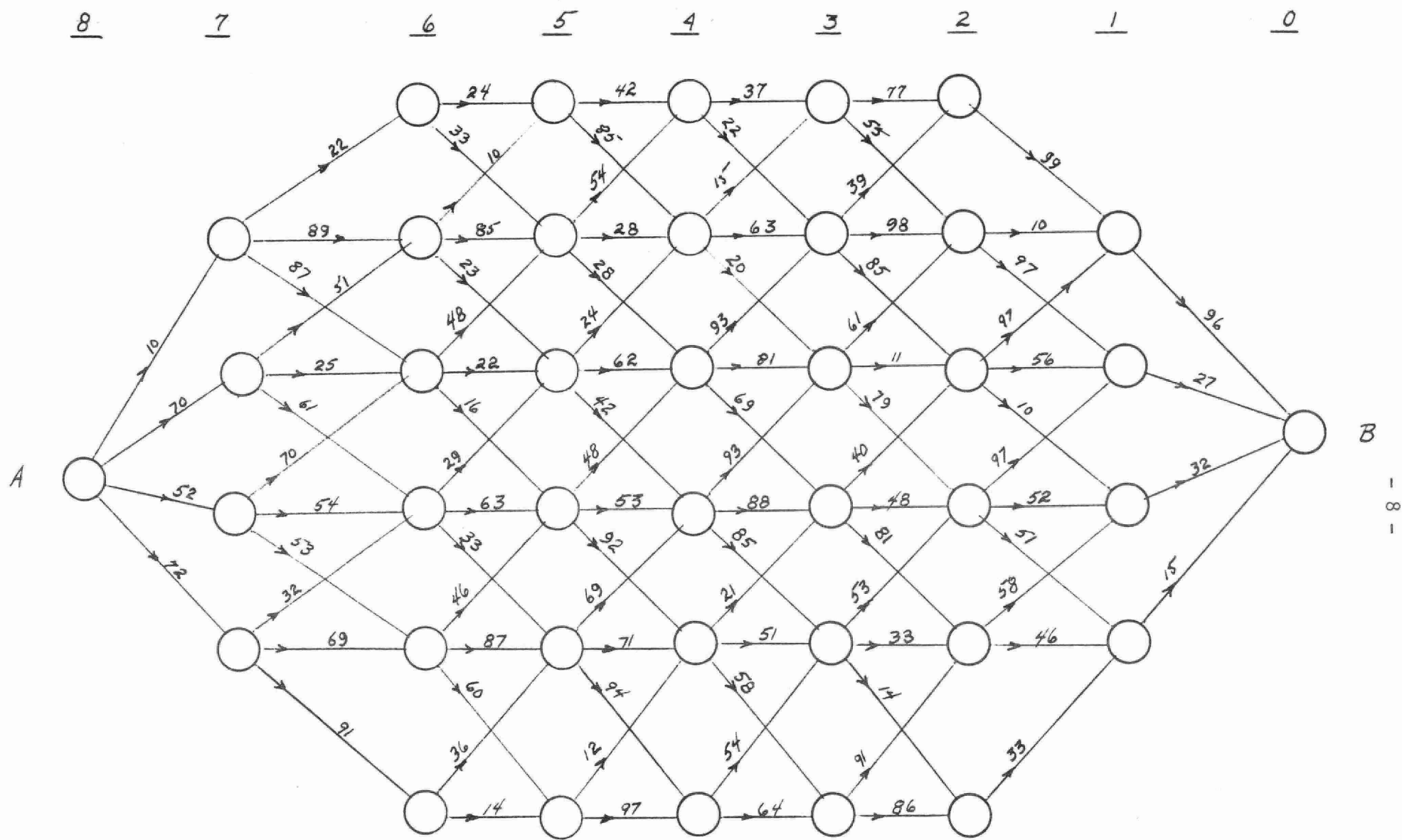


Figure 4. Process Alternatives

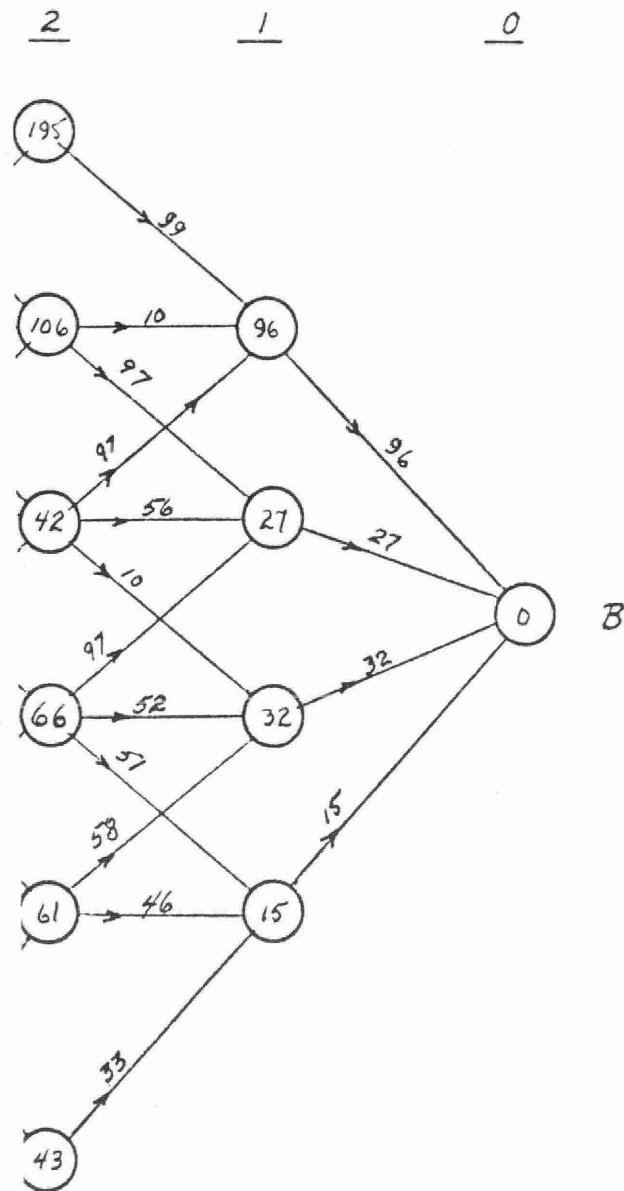


Figure 5. Initialization of Dynamic Programming Procedure

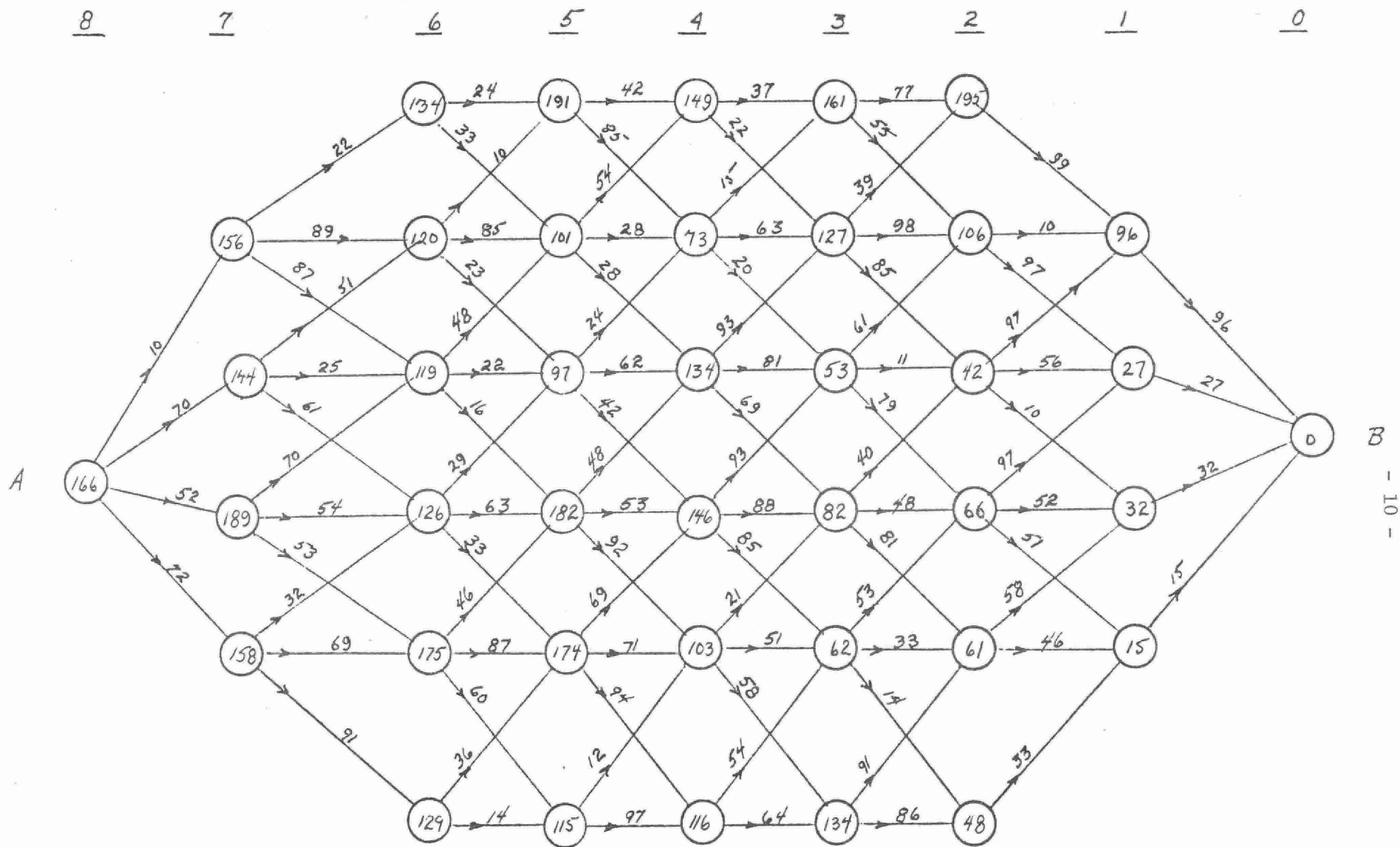


Figure 6. Evaluation of Trajectories

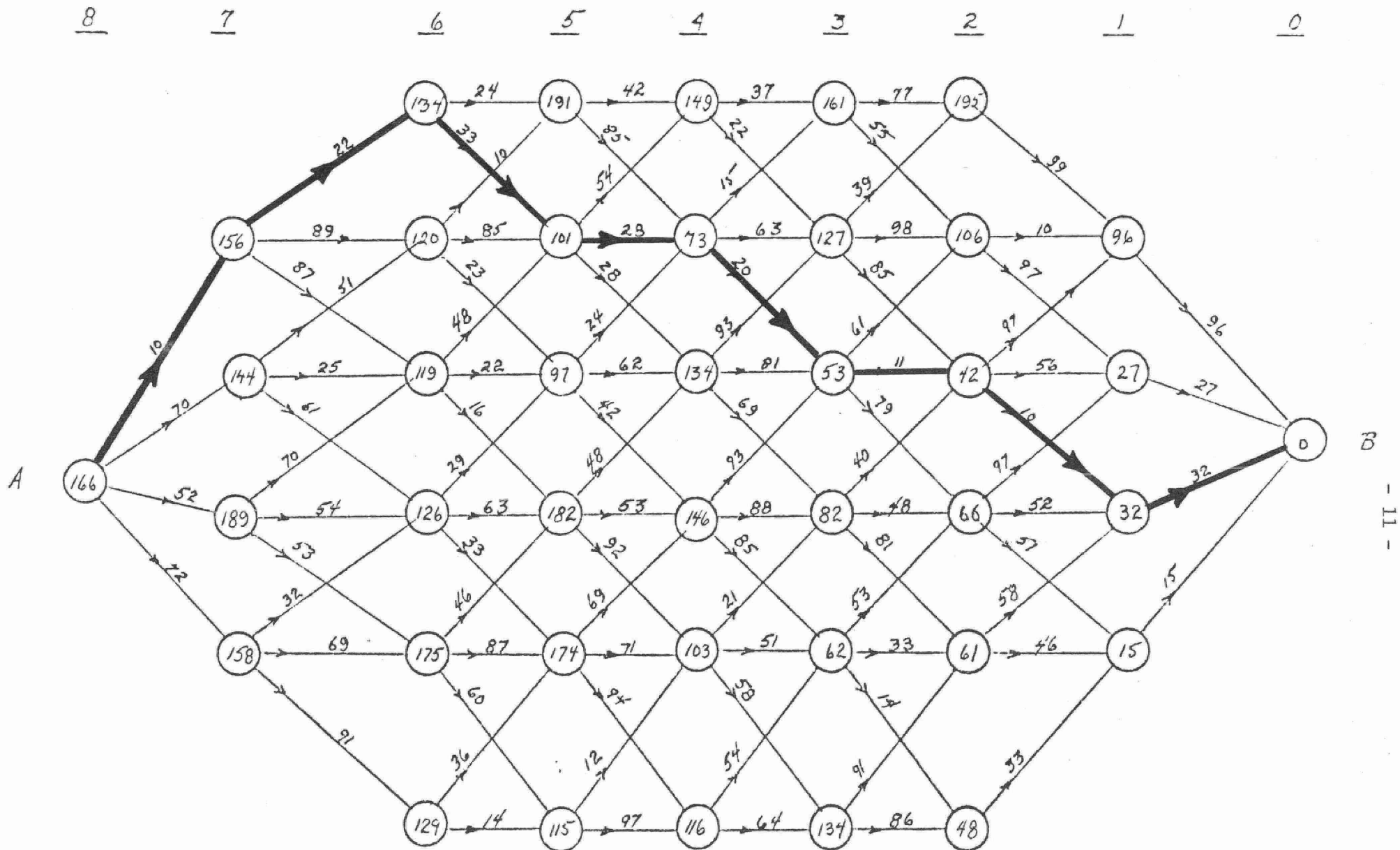


Figure 7. Optimal Trajectory Path

5. Minimization of Pipeline Pumping Energy for Water Transmission Systems

An interesting application of dynamic programming is found in minimizing pumping energy of a pipeline system. Figure 8 is a simplified flow diagram of a series of pumping stations and the corresponding pressure distance profile.

Assume that a flow Q is to be delivered from an inlet suction pressure $(P_s)_1$ by a series of pumping stations L_i miles apart to a terminal at pressure P_t . At the booster stations sufficient energy is added to Q to overcome friction losses in the pipeline as well as the elevation heads.

The pressure balance at the i^{th} station is:

$$(P_d)_i - (P_s)_i = \Delta P_i (L_{i+1} - L_i) + K(E_{i+1} - E_i) \quad (5.1)$$

where ΔP_i = friction pressure drop/unit length
between stations i and $i-1$

$L_i - L_{i-1}$ = the length of line between station i and $i-1$

$E_i - E_{i-1}$ = the difference in elevation between
stations i and $i-1$

K = a proportionality constant

Knowing the initial pressure $(P_s)_1$ and the terminal pressure P_t , optimization is executed so that the sum of the energy requirements over N stages is a minimum. This means finding the values of $(P_s)_i$ and $(P_d)_i$ over the N stages.

Define:

$$Y_i = \text{the pressure boost across the } i^{th} \text{ pumping station} \\ = (P_s)_i - (P_d)_i$$

$$x = \sum_{i=1}^N Y_i = \text{the cumulative pressure boost across the} \\ N \text{ pumping stations}$$

$f_N(x)$ = the minimum energy to deliver Q units/time through the N pump stations when the sum of the cumulative pressure drop across the N stations is x , using an optimal policy.

$g_N(Y_N)$ = the energy for pumping Q at station N .

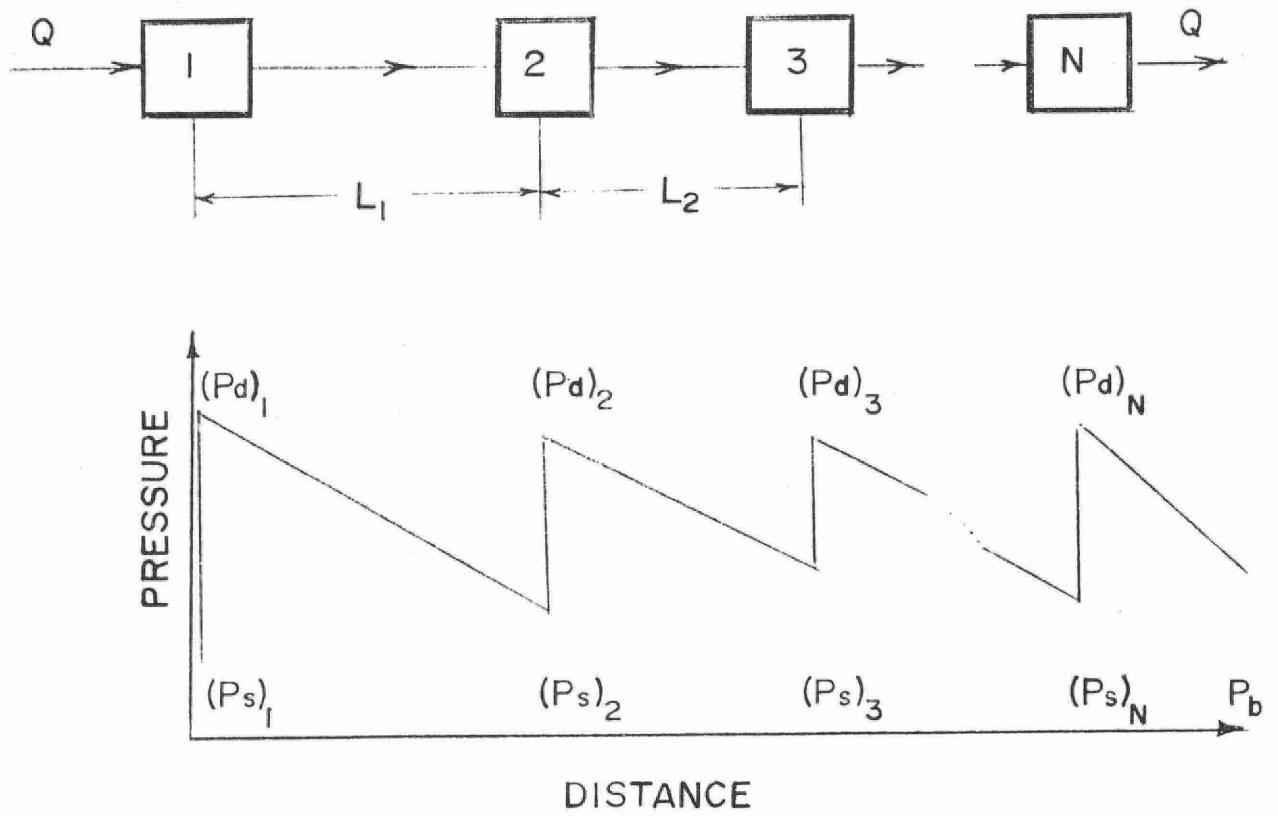


Figure 8. Pipeline Pressure Drop-Distance

The functional equations for dynamic programming are:

$$f_N(x) = \min [g_N(Y_N) + f_{N-1}(x-Y_N)] \quad (5.2)$$

$$0 \leq Y_N \leq x$$

$$f_1 = \min [g_1(Y_1)] = g_1(x) \quad (5.3)$$

$$0 \leq Y_1 \leq x$$

At each stage of the process, there are constraints on the suction and discharge pressures:

$$(P_s)_i \min \leq (P_s)_i \leq (P_s)_i \max; \quad i = 1, 2, \dots, N \quad (5.4)$$

$$(P_d)_i \min \leq (P_d)_i \leq (P_d)_i \max; \quad i = 1, 2, \dots, N \quad (5.5)$$

Execution of equations (5.2) and (5.3) must honour the constraints (5.4) and (5.5).

The optimization may be carried out in many ways depending on the physical situation in the pipeline. Additional data are necessary such as the pump efficiencies, pipe diameters, friction factors, etc. Minimization may be affected by considering a variety of impeller sizes for each pump. In general, the procedure is similar to that of the example in section 4 except a greater number of decisions must be made at each stage.

6. Multi-Unit Water Treatment Plant

A water treatment plant consists of a sequence of units made up of filters, precipitators, chemical mixers, reactors, adsorption beds, etc. The effluent from one unit becomes the feed for succeeding unit. The entire complex may be considered as a multistage process in which the goal is to minimize energy, minimize several types of effluent contaminants or impurities, and minimize the quantity of chemical additives used.

Consider a general type of system where two feed streams F_A and F_B enter unit r , and effluent streams $F_{1,E}$, $F_{N,E}$, and $F_{N-1,E}$ leave units 1, N and $N-1$ respectively as in Figure 9.

Define the following terms:

$F_{i,i-1}$ = the flow rate leaving the i^{th} unit and going to the $(i-1)th$ unit.

$(x_k)_{i,i-1}$ = the concentration of component k in the stream $F_{i,i-1}$; $k = 1, 2, \dots$

$(S_k)_{i,i+1}$ = separation or removal factor for the kth component in the effluent stream $F_{i,i+1}$, mg/ℓ of k /total mg/ℓ entering unit i ; $k = 1, 2, \dots$

$(S_k)_{i,i-1}$ = separation or removal factor for the kth component in effluent stream $F_{i,i-1}$, mg/ℓ of k /total mg/ℓ entering unit i ; $k = 1, 2, \dots$

$(v_m)_i$ = the process variables, temperature, pressure, pH, etc., in the i^{th} unit.

g_i = the performance of the i^{th} unit, involving contaminant reduction and operating energy.

The separation or removal factors for the i^{th} unit may be evaluated as function of the entering streams, their compositions, and the process variables.

$$[(S_k)_{i,i+1} = S(F_{i-1,i}, F_{i+1,i}, (x_k)_{i+1,i}, (x_k)_{i-1,i}, (v_1)_i, \dots, (v_m)_i)] \quad (6.1)$$

$$[(S_k)_{i,i-1} = S(F_{i-1,i}, F_{i+1,i}, (x_k)_{i-1,i}, (x_k)_{i+1,i}, (v_1)_i, \dots, (v_m)_i)] \quad (6.2)$$

Similarly, equations for mass balances can be made around the rth unit with feeds F_A and F_B , around unit $N-1$ with product stream F_C and around units 1 and N . Component balances must be written for any chemical reactions using the separation factors.

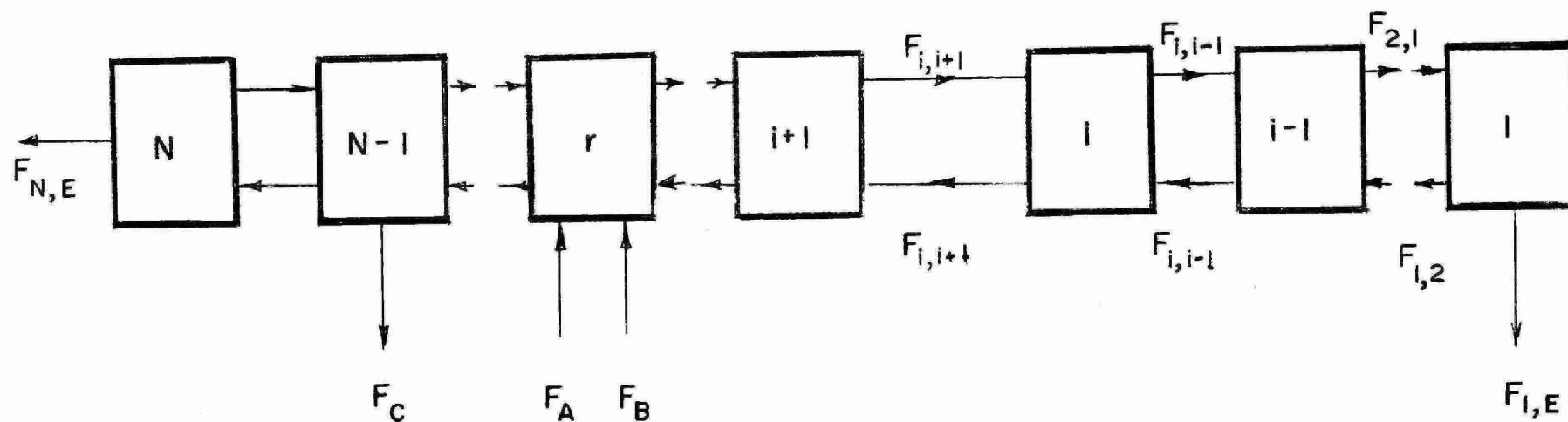


FIGURE 9. MULTI-UNIT WATER TREATMENT PLANT

Thus, the functional equation for the stage-one unit is:

$$f_1[F_{2,1}, (x_1)_{2,1}, \dots, (x_k)_{2,1}] = \min_{(v_m)} \sum_{k=1}^k g_1(F_{2,1}, (S_k)_{1,E}, (S_k)_{1,2})$$

$$k = 1, 2, \dots; m = 1, 2, \dots \quad (6.3)$$

For a two stage process the functional equation is:

$$f_2[F_{1,2}, F_{3,2}, (x_1)_{1,2}, \dots, (x_k)_{1,2}, (x_1)_{3,2}, \dots, (x_k)_{3,2}]$$

$$= \min_{(v_m)_2} \left[\sum_{k=1}^k g_2(F_{2,1}, F_{3,2}, (S_k)_{2,3}, (S_k)_{2,1}) + f_1 \right]$$

$$k = 1, 2, \dots; m = 1, 2, \dots \quad (6.4)$$

$$k = 1, 2, \dots; m = 1, 2, \dots \quad (6.5)$$

In a similar manner the functional equations may be developed for the entire N stage process.

The separation equations as well as the material balance equations are used in conjunction with the functional equations to evaluate the minimum performance index (see Appendix A).

7. Discussion and Recommendations

Conservation of energy has led to a rethinking of design procedures for water and wastewater treatment systems. This added dimension in design is another undetermined variable which must be minimized in addition to cost and effluent quality. The use of dynamic programming allows a sequential stagewise optimization of a very complicated system in which energy should now be considered as an additional criterion in design.

The main advantage to dynamic programming lies in its sequential stage-by-stage character of computation. Of further importance is the fact that constraints on either the control or state variables actually simplify the computation rather than make it more difficult since the discretized range on the variables is reduced.

There are, of course, a number of problems associated with the use of dynamic programming. One of these has been called by Bellman the "menace of the expanding grid". This phenomenon is due to the generation of inadmissible states at the end of the state's discretization; as the calculation propagates, this may lead to many inadmissible values. The second problem, and the most serious one, results from the dimensions of the state variables. With a large number of state variables which are discretized at each stage of dynamic programming, a computer storage problem may result. This has been called by Bellman "the curse of dimensionality".

These problems are trivial in the light of the advantages and simplicity of dynamic programming. Careful formulation of the minimization procedure can eliminate them.

As a tool in making optimal decisions for energy allocation and utilization, dynamic programming is recommended for future submissal of design proposals wherein several alternatives are considered as in Figure 3 and some sort of performance index is minimized with respect to operating costs, effluent specifications and energy requirements.

Further reductions in energy requirements may be achieved by utilization of more recent developments in the unit operations design as well as innovative treatment systems. This will result in a wider range of alternatives which will aid the approach to energy conservation.

REFERENCES

1. Athans, M. and Falb, P.L., "Optimal Control", McGraw-Hill Book Co., 1966.
2. Bellman, R., "Dynamic Programming" Princeton University Press, 1957.
3. Bellman, R., "Applied Dynamic Programming", Princeton University Press, 1962.
4. Lapidus, L., and Luus, R., "Optimal Control of Engineering Processes" Blaisdell Publishing Company, 1966.
5. Zarnett, G.D., "Energy Requirements for Conventional and Advanced Wastewater Treatment", Publication No. W47, Applied Sciences Section, Pollution Control Branch, Ministry of the Environment, Toronto, October 1975.
6. Zarnett, G.D., "Energy Requirements for Wastewater Treatment Equipment", Technical Note No. TN 7008, Applied Sciences Section, Pollution Control Branch, Ministry of the Environment, Toronto, July 1976.
7. Zarnett, G.D., "Energy Requirements for Water Treatment Systems", Research Paper No. S2043, Applied Sciences Section, Pollution Control Branch, Ministry of the Environment, Toronto, November 1976(a).

APPENDIX I

EXAMPLE OF MATHEMATICAL OPTIMIZATION OF A TREATMENT SYSTEM BY DYNAMIC PROGRAMMING

Consider a system with P discrete pollution treatment steps. If we consider a typical stage k , as in Figure A1, $x(k)$ is the input concentration of wastewater contaminant and $x(k+1)$ is the output concentration of stage k . Also, let $u(k)$ be the energy input at stage k . Assume that we have sufficient knowledge of the system so that we have a system equation for any stage;

$$x(k+1) = ax(k) + ku(k)$$

with initial feed concentration known, as $x(0)$, and $k = 0, 1, 2, \dots$; and, a and b are constants.

To optimize P stages of the system we must have a means to evaluate the performance. This is usually done using a performance index. We can formulate a performance index by saying that we wish to minimize the contaminant output and at the same time minimize the energy requirement. To penalize ourselves heavier, we should minimize the square of the concentration and the energy. If I is a performance index, then:

$$I = x^2(k) + u^2(k-1)$$

Because we must minimize P stages, the performance index is of the form

$$I[x(0), P] = \sum_{k=1}^P [x^2(k) + u^2(k-1)]$$

Thus, the problem is to find $u(0), u(1), \dots, u(P-1)$ such that $I[x(0), P]$ is minimized.

The solution using dynamic programming is as follows: The final state $x(P)$ is taken to be the origin. First consider one stage, with $P=1$, and go from $x(0)$ to $x(1)$ by a selection of $a(0)$ which minimizes $I[x(0), 1]$.

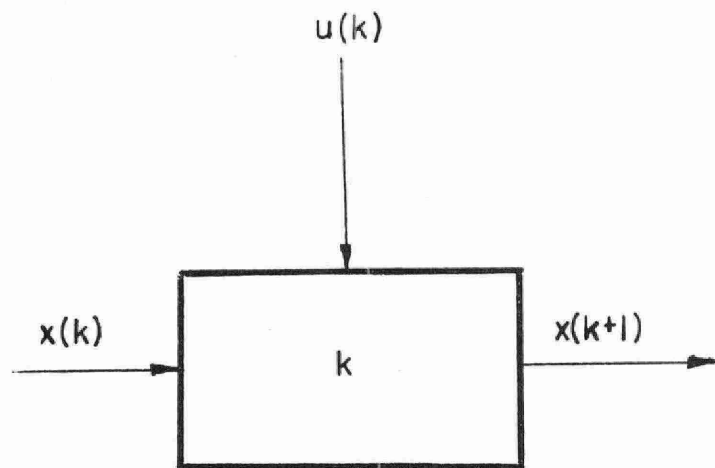


FIGURE A1

Here:

$$I[x(o),1] = x^2(1) + u^2(o)$$

But, it is noted that the origin can be reached in one step, $x(1)=0=\text{origin}$.

$$I[x(o),1] = u^2(o)$$

For equation (4.1) to be satisfied, it is necessary that

$$u(o) = \frac{x(1) - ax(o)}{b} = \frac{a}{b} x(o)$$

Note that this value of $u(o)$ is a special case since it is independent of the performance index. Thus, there is only one unique value of energy $u(o)$, which connects $x(o)$ to $x(1)$ in one step. Knowing $u(o)$

$$I[x(o),1] = \frac{a^2}{b^2} x^2(o)$$

Next, consider $P=2$ and find $u(o)$ and $u(1)$ which take the system from $x(o)$ to $x(2)$ while minimizing $I[x(o),2]$. To keep the nomenclature consistent, $x(2) = 0 = \text{origin}$, and:

$$I[x(o),2] = x^2(1) + u^2(o) + x^2(2) + u^2(1)$$

Figure A2 shows a schematic of the steps for $P=1$ and $P=2$. Now

$$x(2) = 0 = ax(1) + bu(1)$$

from which:

$$u(1) = - (a/b) x(1)$$

and thus:

$$I[x(o),2] = x^2(1) + u^2(o) + u^2(1) = (1 + \frac{a^2}{b^2})x^2(1) + u^2(o)$$

To obtain the extremum of $I[x(o),2]$ for a choice of $u(o)$, we can differentiate $I[x(o),2]$ and set the results equal to zero.

$$I[x(o),2]/u(o) = 0 = 2u(o) + 2(1 + \frac{a^2}{b^2})x(1) \partial x(1)/\partial u(o)$$

Since:

$$x(1) = ax(o) + bu(o) \quad \partial x(1)/\partial u(o) = b$$

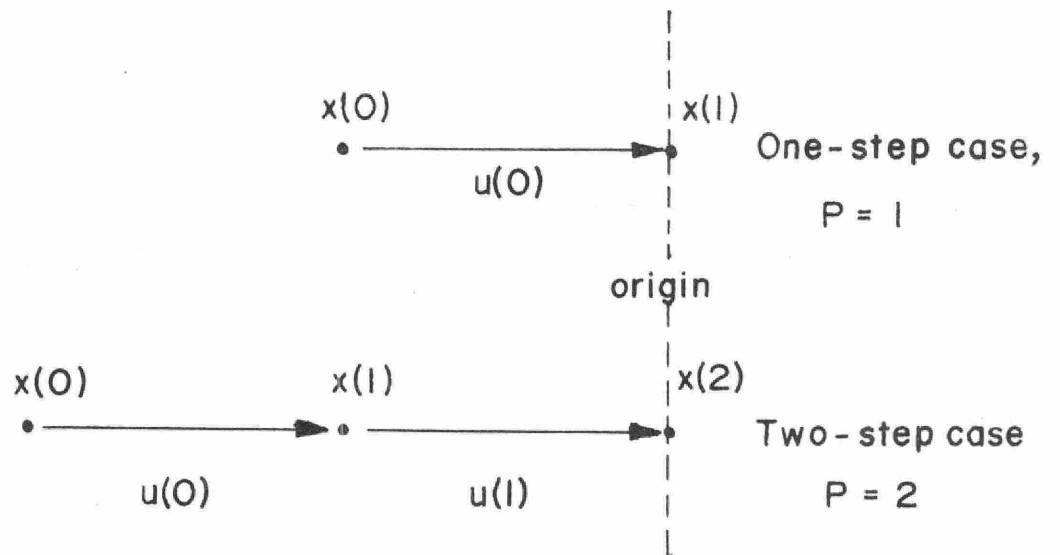


FIGURE A2. TWO STEPS IN DYNAMIC PROGRAMMING EXAMPLE

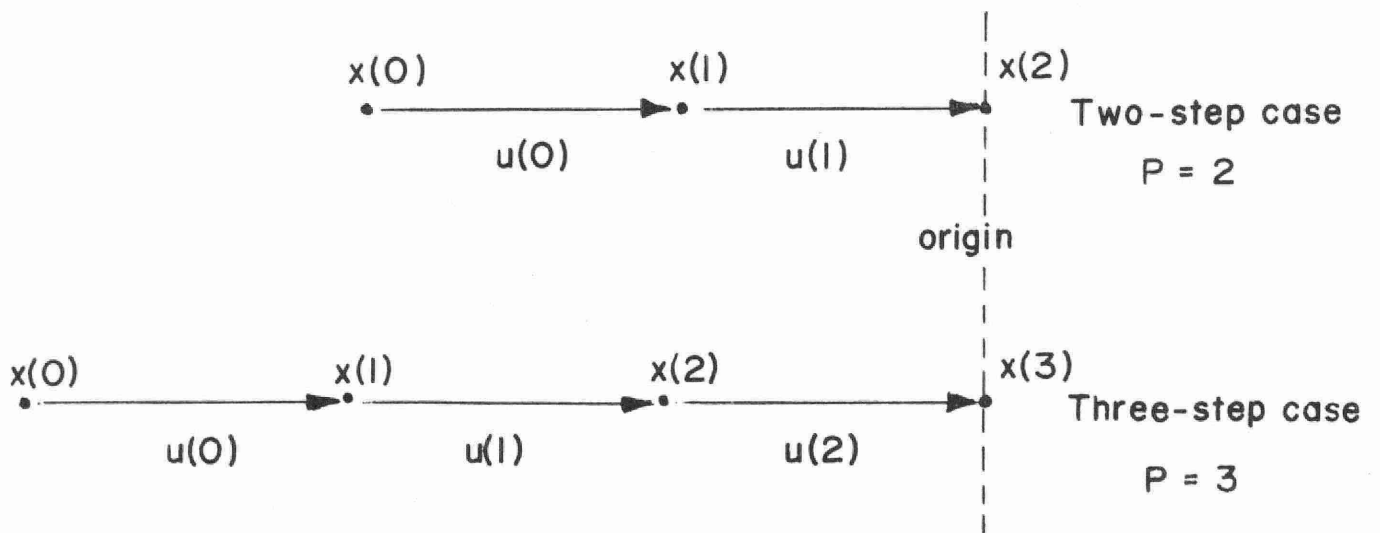


FIGURE A3. THREE STEPS IN DYNAMIC PROGRAMMING EXAMPLE

Hence:

$$\partial I[x(o), 2] / \partial u(o) = 0 = 2u(o) + 2(1 + \frac{a^2}{b^2}) [ax(o) + bu(o)]b$$

or

$$u(o) = - \frac{ab + (a^3/b)}{1 + a^2 + b^2} x(o)$$

Thus $x(1)$, $u(1)$ and $x(2)$ may be obtained. We can go on to a third stage to get $x(o)$ to $x(3)$ by a proper selection of $u(o)$, $u(1)$, and $u(2)$ while $I[x(o), 3]$ is minimized. In this case $x(3) = 0 = \text{origin}$, and

$$I[x(o), 3] = x^2(1) + u^2(o) + x^2(2) + u^2(1) + u^2(2)$$

Figure A3 shows the three stage process. This process may be continued for P stages.



96936000009319